

Interpolación de Newton

Introducción.

Es un método de interpolación polinómica. Aunque sólo existe un único polinomio que interpola una serie de puntos, existen diferentes formas de calcularlo. Este método es útil para situaciones que requieran un número bajo de puntos para interpolar, ya que a medida que crece el número de puntos, también lo hace el grado del polinomio.

Realice la interpolación de Newton para $x = 4$ de los valores de la tabla siguiente:

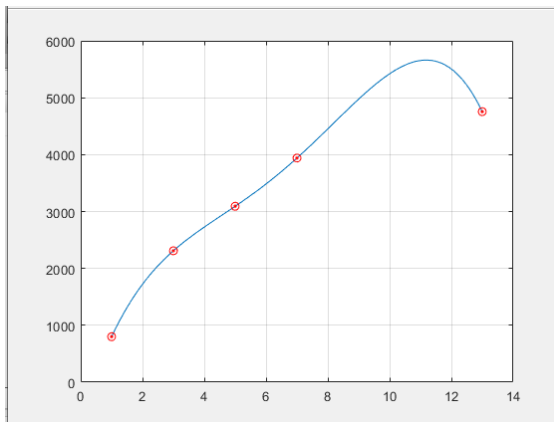
función

| x | f(x) |
|----|------|
| 1 | 800 |
| 3 | 2310 |
| 5 | 3092 |
| 7 | 3940 |
| 13 | 4755 |

>> Inter_new

Ingrese el vector fila x:[1 3 5 7 13]

Ingrese el vector fila y:[800 2310 3092 3940 4755]



Polinomio interpolacion Newton:

$$800+755*(x-1)-91*(x-1)*(x-3)+16.5417*(x-1)*(x-3)*(x-5)-1.7474*(x-1)*(x-3)*(x-5)*(x-7)$$

>> x=4

$$>> Fx=800+755*(x-1)-91*(x-1)*(x-3)+16.5417*(x-1)*(x-3)*(x-5)-1.7474*(x-1)*(x-3)*(x-5)*(x-7)$$

$$Fx = 2726.6$$

Código fuente usado:

```
%Ingrese los vectores flia
x=input('Ingrese el vector fila x:');
y=input('Ingrese el vector fila y:');

xa=x;ya=y;

%Formacion de las diferencias divididas
d=zeros(length(y));
d(:,1)=y';
for k=2:length(x)
    for j=1:length(x)+1-k
        d(j,k)=(d(j+1,k-1)-d(j,k-1))/(x(j+k-1)-x(j));
    end
end

%Formacion del polinomio
for w=1:length(x)
    ds=num2str(abs(d(1,w)));
    if w>1
        if x(w-1)<0
            sg1='+';
        else
            sg1='-';
        end
    end
    if d(1,w)<0
        sg2='-';
    else
        sg2='+';
    end
    if w==1
        acum=num2str(d(1,1));
    elseif w==2
        polact=['(x' sg1 num2str(abs(x(w-1))) ') '];
        actual=[ds '*' polact];
        acum=[acum sg2 actual];
    else
        polact=[polact '.*' '(x' sg1 num2str(abs(x(w-1))) ') '];
        actual=[ds '*' polact];
        acum=[acum sg2 actual];
    end
end
end

%Presentacion de resultados
fprintf(' n Valores de X y Y n ');
%disp(xa);
%disp(ya);
fprintf('Polinomio interpolacion Newton: %s n')
disp(acum)
%x=input('\n X interp = ');
if x>max(xa)|x<min(xa)
    fprintf('t Punto fuera de rango. El resultado puede equivocado n');
```

```

end
xinterp=x;
yinterp=eval(acum);
fprintf(' Y(%g) = %g n',x,yinterp);

%Grafica de los puntos

fprintf('\n Pulse cualquier tecla para ver la grafica de los puntos n');
%pause
xg=linspace(min(xa),max(xa));
x=xg;yg=eval(acum);
plot(xg,yg,xa,ya,'.r',xinterp,yinterp,'or');
grid

```

Conclusión

Existen ciertas ventajas en el uso de este polinomio respecto al polinomio interpolador de Lagrange. Por ejemplo, si fuese necesario añadir algún nuevo punto o nodo a la función, tan sólo habría que calcular este último punto, dada la relación de recurrencia existente y demostrada anteriormente.

Sanchez González Oscar Eduardo.

ingeniería en Sistemas Computacionales.

Métodos numéricos

NUA:304987